社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

[Encouragement Talk] Evaluation of Robustness and Adaptability of a Biologically-inspired MANET Routing Protocol

Narun ASVARUJANON†, Kenji LEIBNITZ†, Naoki WAKAMIYA†, and Masayuki MURATA†

† Advanced Network Architecture Laboratory, Department of Information Networking,
Graduate School of Information Science and Technology, Osaka University,
1-5 Yamadaoka, Suita, Osaka 565-0871, Japan
E-mail: †{narun-a,leibnitz,wakamiya,murata}@ist.osaka-u.ac.jp

Abstract Mobile ad hoc networks (MANETs) have various merits over a traditional wired network, e.g. requiring no fixed infrastructure. However, routing in MANETs faces many difficulties, e.g. frequent topology changes and easily interfered multiple access medium. Therefore, a routing protocol is required to be robust and adaptive against topology changes and packet collisions. Biologically-inspired systems are known for their robustness and self-adaptability to a changing environment. Therefore, our proposed protocol (MARAS) utilizes a biologically-inspired mechanism, called *attractor selection*, to achieve robustness and adaptability. In MARAS, each node sets up a routing vector toward its destination. The routing information is continuously updated by the path condition information, called *activity*, and used in the next hop selection process of data packet forwarding. The evaluation results show that our routing protocol has a better performance than AODV in both static and random failure scenarios. In particular, MARAS has higher delivery efficiency with lower transmission overhead per successfully delivered packets.

Keywords Biologically-inspired networking, Routing protocol, Ad hoc networks

1 Introduction

Mobile Ad Hoc Network (MANET) is the research domain that has been receiving a lot of attention in the last few decades. MANETs differ from the traditional wired networks because they are independent of a fixed infrastructure; this allows the mobile nodes to move freely and makes many useful applications possible in MANETs. However, this flexibility causes difficulties in routing as it comes with unpredictable dynamic changes in topology, decentralized control, limited energy, and limited bandwidth [1]. For this reason, one of the most active research areas in MANETs is on routing protocols.

Many MANET routing protocols have been proposed in the past and they can be distinguished into two main categories: *proactive* and *reactive* (or *on-demand*) protocols. We focus our research on on-demand protocols as proactive methods consume too much energy in exchanging the routing information on a periodic basis, especially pure proactive protocols like DSDV. In addition, it has been shown that DSR (reactive) is better than OLSR (proactive) in terms of energy consumption [2]. Among the on-demand protocols, we choose AODV as it is more scalable and more adaptive than DSR [3]. However, AODV has its own weaknesses, i.e., it causes high load on the network because of routing overhead (mainly flooding) and it does not take the link qualities into account, which possibly results in selecting unstable links [4]. Therefore, we aim at designing a routing protocol, which is more robust and adaptive against unstable conditions in the network and causes lower overhead routing protocol than AODV.

To achieve robustness and adaptability, we consider a biologically-inspired mechanism. As biological systems are well-known for their robustness and adaptability, there is a lot of research adopting mechanisms inspired by biology, e.g., swarm intelligence and ant colony optimization (ACO). For MANETs, many biologically-inspired routing protocols have been proposed and most of them are based on swarm intelligence, e.g., AntHocNet [5] and BeeAdHoc [6]. Note that our protocol however uses a biologically-inspired mechanism from cell biology called *attractor selection* and is not based on swarm intelligence.

Our robust and adaptive mobile ad hoc routing with attractor selection (MARAS) is an extended work from [7] which is based on [8,9]. MARAS is a noise-driven on-demand protocol which uses feedback of delivered data packets from the destination for route maintenance. Using the feedback information along with the attractor selection mechanism allows MARAS to recover from link failures without issuing any additional broadcast control message like AODV. According to the evaluation results, in most scenarios MARAS has higher delivery efficiency than AODV and in high node density cases MARAS has lower transmission overhead per successfully delivered packet.

This paper is organized as follows. First, we introduce the attractor selection mechanism and the derived mathematical model in Section 2. Next, we describe our protocol in Section 3. Then, in Section 4, the evaluation results are presented and discussed. Finally, we conclude and list future work in Section 5.

2 Background and Model

In this section, we introduce the background of the adopted biologically-inspired mechanism and our derived mathematical model. Additionally, we explain the notation that will be used in the rest of this paper.

2.1 Attractor Selection Mechanism

The attractor selection mechanism is modeled after the behavior of *E. coli* cells, which is capable of adapting to dynamically changing nutrient conditions in the environment without an embedded rule-based mechanism [10]. The equilibrium conditions in the metabolic network are called *attractors*. Since the biological systems are dynamic, there are changes in the system all the time. For example, when the cell becomes unstable by external influences or internal noise, its gene expression state will be driven to other attractors to return the cell to a stable condition. As there are more than one possible stable conditions, there is a mechanism to select a suitable attractor among multiple attractors, which is called attractor selection. Please refer to [7] for more details.

2.2 Our Mathematical Model

The attractor selection is adopted in our protocol for next hop selection among neighbors. Hence, we map the vector of neighbors to $\vec{m}$, which contains value m_i, called *state value*, indicating if the neighbor i^{th} should be selected and map activity α to the information which shows the goodness of the current routing condition. Moreover, as we consider unicast traffic between the source and the destination, the selection shall select a single next hop neighbor at a time. Therefore, we design the attractor selection function as

$$\frac{dm_i}{dt} = \frac{s(\alpha)}{1 + m_{max}^2 - m_i^2} - d(\alpha)m_i + \eta_i, \qquad (1)$$

where $i = 1, \ldots, M$, $m_{max} = \max_{j=1,\ldots,M}(m_j)$, $s(\alpha) = \alpha[\beta\alpha^\gamma + \varphi^*]$, $d(\alpha) = \alpha$, $\varphi^* = 1/\sqrt{2}$, and η_i is the white noise.

In the case that the activity α is high, the Eqn. (1) gives

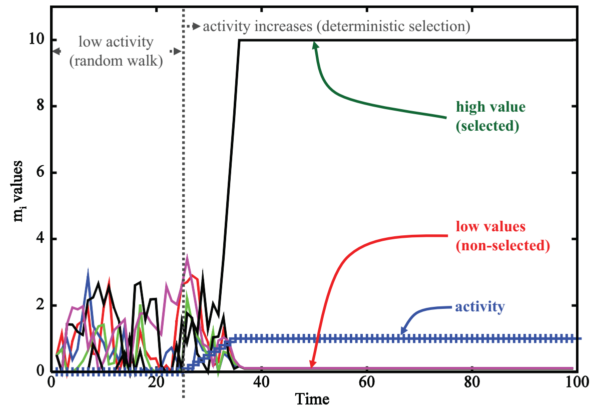


Fig. 1: The dynamics of M alternatives' value from attractor selection model ($M = 6$). The solid lines represent the m_i values while '+' line represents the activity α.

the $\vec{m}$ which has a single high value and $M-1$ low values. This means that only one neighbor will be selected as the next hop as only the maximum value is selected in our protocol. While in the case that activity α is low, the Eqn. (1) gives a random $\vec{m}$ where each member m_i has roughly the same value, so that the appropriate selection can be found easily requiring only small differences.

The dynamics of M alternative values from Eqn. (1) is shown in Figure 1. From the time $t = 0$ to 25, the α is low, therefore, each value m_i receives more effect from noise and has a random value. When the solution is found, i.e., after time $t = 26$, α starts increasing. Therefore, the gap between selected value and non-selected values grows larger and becomes stable with one high value and $M-1$ low values once the $\alpha = 1.0$ which indicates that the system reaches the suitable attractor.

3 Biologically-inspired Routing Protocol for MANETs

MARAS is an on-demand routing protocol which sets up the route upon request. In MARAS, each node maintains its own routing table and neighbor list. We assume the bidirectional connectivity between each pair of neighbor nodes. MARAS uses feedback packets to update the routing information and ignores the outdated information. As the unidirectional links will never be updated by the feedback packets, they are automatically ignored.

3.1 Route Establishment

We use the same route establishment mechanism as stated in [7] which utilizes broadcast route request packet (RREQ) and unicast route reply packet (RREP). The state value of a neighbor in routing vector is set to one when the RREP is received from that neighbor while all others are 0. Moreover, if the current node has no route entry for that destination, then it will set up a new random vector which contains equal state values $m_i = \lambda$ for every neighbor i and starts the random walk mechanism.

3.2 Routing Information

The routing information stored at each node in the route entry are 1) a destination address, 2) an attractor selection vector, called a *routing vector* $\vec{m} = (m_1, m_2, \ldots, m_M)$, 3) an activity α, 4) a precursor list, and 5) a feedback window.

Regarding the precursor list, it contains the addresses of the source nodes which use this route entry along with the last neighbor that sent the data packet originated at that source node via the current node. The feedback window will be explained in Section 3.4.1.

3.3 Data Packet Forwarding

Using attractor selection, MARAS selects the neighbor which has the *maximum state value* in the routing vector as a next hop. The data packet is forwarded to this next hop and the process repeats itself until it reaches the destination. The next hop is selected by the maximum state value as it shows the highest potential of that neighbor on delivering the data packet to the destination.

The concept of attractor selection along with the maximum state value favors the next hop selection in a way that, MARAS will keep selecting the same next hop as long as the activity is high. When the activity drastically decreases, the noise will increase the other candidates' state values to allow the selection of a different neighbor. Hence, MARAS is able to quickly recover from the undesirable conditions.

3.4 Route Maintenance

MARAS maintains the same route as long as it is being used and removes unused route entries after a period of time to save the resources. In order to keep the routing information up-to-date, MARAS uses the feedback packet to learn the current condition of the network. Moreover, it updates the routing vector using a calculated activity to adapt the next hop selection according to the current network condition. The activity is decayed over time due to the reasons stated in [7]. Regarding the local connectivity maintenance, we use HELLO packet which is similar to AODV [11].

3.4.1 Feedback Packet

Upon the data packet arrival at the destination, a feedback packet is generated and sent back to the source. The feedback packet exploits the *memorized previous hop* in the precursor list at each intermediate node to take the most recent route back to the source and avoid getting lost. During its journey, it leaves its travelled hop count information in each intermediate node's *feedback window* for the purpose of activity calculation. The feedback window is the sliding window which keeps the hop count to destination and deletes this hop count after *window interval* T to avoid using the outdated information.

3.4.2 Activity Calculation

The activity of each routing vector is calculated upon the feedback packet arrival based on the most recent feedback packet's travelled hop count and the minimum travelled hop count in the feedback window. Supposed that the feedback packet arrives at time t, the activity is calculated using the following equation:

$$\alpha(t) = \frac{\min_{\forall w_k \in W} w_k}{w_n}, \tag{2}$$

where W is the feedback window which contains n hop count values, each of which is $w_k \in W$ for $k = 1, \ldots, n$, and w_n is the travelled hop count of the most recent feedback packet arriving at time t.

This activity changes according to the hop count to the destination in the range between 0 and 1. If the hop count to the destination becomes larger, then it means that the current path to the destination is unstable, e.g., link failure or node movement occurs, and the attempt to find a better path should be made. Therefore, the activity will decrease in such situation and the effect from noise will induce a random walk. On the other hand, once a shorter path is found, the $\alpha(t)$ will immediately become 1, and MARAS will keep using this path until another change occurs in the network.

3.4.3 Routing Vector Update and Activity Decay

In our protocol, we use the simple activity decay equation on the stored activity:

$$\alpha_{decayed} = \alpha_{stored} - \delta, \tag{3}$$

where the decay constant $\delta = 0.1$ is used for the current implementation. The decay process is periodically performed over interval τ. The activity decay mechanism is performed regardless of the feedback packet arrival. Therefore, when there is no incoming feedback packet, the activity will continuously be decayed and the routing vector will be updated by using the decayed activity.

To keep the information in the routing vector consistent to the value of activity, the routing vector is always updated after there is any change of the activity value, i.e., on feedback packet arrival and activity decay.

4 Evaluation

We evaluate MARAS by performing simulations with a network simulator called QualNet. We compare MARAS to AODV in QualNet version 4.0 which is based on AODV draft 8 [11] with extensions from draft 9 [12].

4.1 Simulation Setting

The area of evaluation scenario is $1500 \times 1500\,\mathrm{m}^2$. Nodes are placed uniformly within this area using the uniform node placement tool available in QualNet. The tool devides the area into grids with the number of tiles equal to a number

Table 1: Simulation parameters of MARAS

Parameter	Value
High value in $s(\alpha)$ calculation β	10
The exponent of α in $s(\alpha)$ calculation γ	3
Window interval T	1.0 s
Initial random-walk vector's state value λ	0.5
Decay constant δ	0.1
Decay interval τ	1.0 s

of node and places the node randomly within the tile. Furthermore, nodes are placed in order from lower left corner to the upper right corner. The number of nodes is varied from 49, 121, 169, to 256 in the same area to study the effect of node density. The node positions remain the same throughout the simulation as node movement is not considered in this evaluation scenario. Instead, we study the adaptability of our proposal by using a failure model which is described in Section 4. 2. Moreover, each point in the evaluation figure is the average value from 100 simulation runs.

Each node in the simulation uses the IEEE 802.11b wireless module with data rate of 2 Mbps. The estimated radio range is 510 s as we use QualNet's free-space model without fading. Regarding the traffic, constant bit rate (CBR) is used as an application with UDP as a transport layer protocol. In order to observe the pure MARAS performance, we selected UDP as a transport protocol to avoid effects from TCP's control mechanisms, e.g., congestion control. We use CBR bit rate of 8 kbps which sends out 10 packets per second. The simulation time is 3000 s where the traffic starts at 0 s and ends at 2500 s, and the last 500-second interval is spared for any delayed packets. Additionally, the wireless interface buffer at each node can store 50,000 packets which could be considered as infinite for the current traffic condition.

In this evaluation, MARAS is compared to 3 different variations of AODV which are *AODV*, *AODV+L*, and *AODV+LI*. First, AODV is a standard AODV configuration without local route repair feature and only the destination can respond to a route request message. Next, AODV+L is a standard AODV with an addition of local route repair feature. Finally, AODV+LI is a standard AODV including local route repair feature and allowing an intermediate node to respond to a route request. We use these variations of AODV to study the effect of the amount of route recovery control messages.

The specific parameters of MARAS are described in Table 1. The other parameters of AODV and MARAS, which are not stated here, are the default values according to QualNet 4.0.

4. 2 Failure Model

In this evaluation, a failure model is used to simulate topology changes which are caused by joining nodes and leaving nodes. We force a number of nodes to fail at the same time by switching their wireless interfaces off using the available API in QualNet. Consequently, the link failures occur and the route recovery performance can be evaluated using this failure model. Failing nodes are randomly selected from all the nodes in the simulation area excluding the source(s) and the destination(s).

To maintain the number of active nodes, the failure period is shortened as the number of failure occurrences is increased. In other words, the number of failure occurrences proportionally reflects the degree of network dynamics.

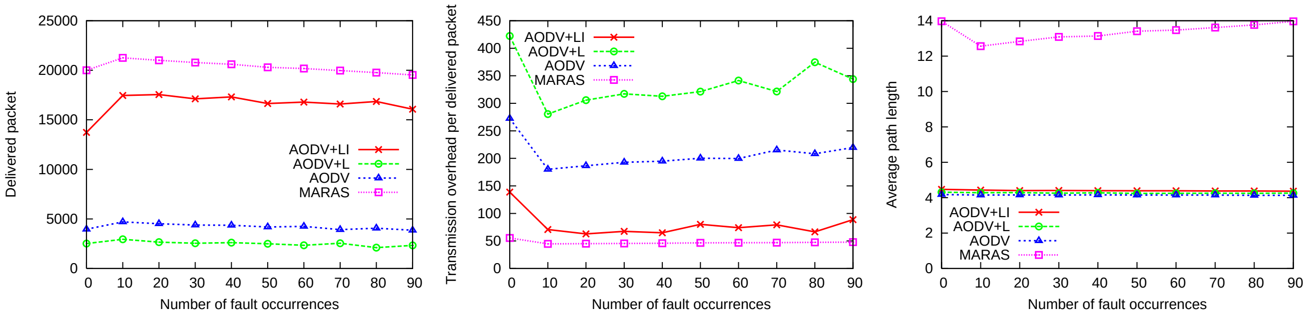
The settings of numbers of failure occurrences are 0 - 90 with the increment step of 10 occurrences. The first group of nodes starts failing at 0 s and the failures iteratively occur every (2,500/the number of failure occurrences) s. The failure lasts for the same interval and the last failure ends at 2,500 s. Note that the value 0 means no failure occurrences or a static scenario. Moreover, the numbers of failing nodes are approximately 25% of all nodes, which are 12, 30, 42, and 64 nodes for 49, 121, 169, and 256 nodes scenario, respectively.

4. 3 Single Session Scenario

In this scenario, we have only one source and destination pair. The source is the first node, which is positioned at the lower left corner, and the destination is the last node, which is positioned at the upper right corner of the scenario area. The results shown in Figure 2 are from the 256 nodes scenario. However, we also show the comparison with other node density values in Figure 3.

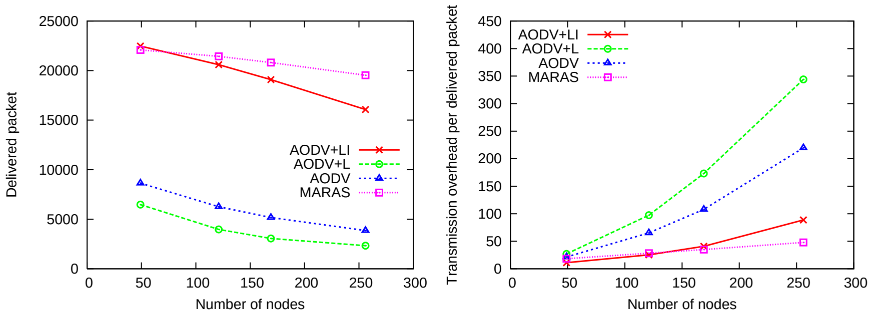
The delivery efficiency results are shown in Figure 2(a) on the Y-axis against the number of failure occurrences on the X-axis. The delivery efficiency is measured from the number of delivered packet at the destination out of 25,000 packets in this simulation. From Figure 2(a), it can be observed that MARAS has the highest delivery efficiency in all cases. Moreover, among the variations of AODV, AODV+LI has the highest delivery efficiency which reflects that the route recovery control messages greatly affect the delivery efficiency.

We define the overhead metric as the transmission overhead per successfully delivered packet. This transmission overhead is the sum of the number of unicast and broadcast transmissions on the network layer from every node in the network. Therefore, this metric indirectly indicates the network load inflicted by the routing protocol per successfully delivered packet. According to Figure 2(b), even though MARAS uses a feedback packet per every successfully delivered packet, which causes traffic to be doubled, MARAS achieves lower overhead in all cases. Moreover, the overhead of MARAS remains almost constant regardless of the num-



(a) Delivery efficiency (b) Transmission overhead (c) Average path length

Fig. 2: Evaluation results of the single session scenario with 256 nodes against the number of fault occurrences



(a) Delivery efficiency (b) Transmission overhead

Fig. 3: Evaluation results of the single session scenario with 256 nodes and 90 failure occurrences against the number of nodes

ber of failure occurrences while the overhead of AODV has the tendency to increase with the network dynamics.

The last metric is the average path length which is calculated by averaging travelled hop count of successfully delivered packets. Figure 2(c) shows the average path length results and MARAS has approximately 3.5 times longer path length than the others. Normally, MARAS should cause more overhead and has lower delivery efficiency as it has a longer path. However, surprisingly, neither the overhead nor the delivery efficiency are the same as the expected results as in normal cases. Our assumption is that, MARAS takes a longer path to avoid using the congested or unstable links unlike AODV which insists on using the shortest path.

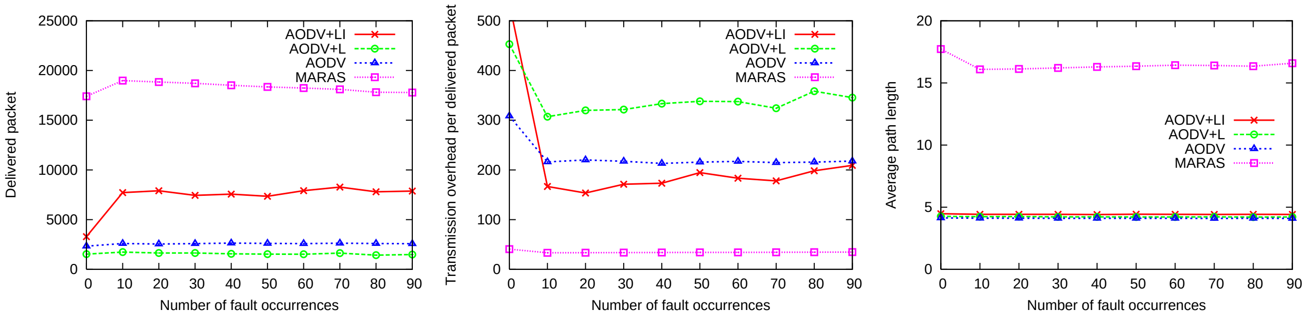
Note that the sudden changes between $x = 0$ and $x = 10$ in every graph are caused by the different number of active nodes. As the failure model puts 25% of nodes into inactive state, there are less collisions caused by HELLO packets and less radio interference. Therefore, improvements of performance are expected in such situation.

In Figure 3, we show the results from 90 failure occurrences scenario (the worst case). Figure 3(a) has the delivery efficiency on the Y-axis and the second row has the overhead on the Y-axis. The X-axis is the number of nodes or node density for both rows. It can be observed that the performance of AODV regarding node density drops in non-linear manner while the performance of MARAS drops in linear manner. Hence, it is enough to say that MARAS is more scalable than AODV.

4. 4 Two Sessions Scenario

After we have observed the average path length of MARAS, we decide to evaluate the performance of MARAS when the traffic increases compared to AODV. This scenario has two source and destination pairs. The first pair is the same to the single scenario while for the second pair the source is the second node and the destination is the second to last node. For the simplicity of comparing the result with the single session scenario, all the results are the average of the two sessions in this scenario.

In Figure 4, similar results to the single session scenario can be seen from 256 nodes scenario. While the average path length of MARAS is still approximately 3–4 times longer than AODV, MARAS has higher delivery efficiency and lower transmission overhead per successfully delivered packet in all cases. Moreover, the performance gap between MARAS and AODV increases, which shows that MARAS can handle a higher amount of traffic than AODV before the performance degrades. Furthermore, another evidence, which shows that AODV inflicts higher network load than MARAS, can be observed from a large difference in delivery efficiency of AODV+LI when comparing the Figure 4(a) and



(a) Delivery efficiency (b) Transmission overhead (c) Average path length

Fig. 4: Evaluation results of the two sessions scenario with 256 nodes against the number of fault occurrences

Figure 2(a), which are both 256 nodes scenarios.

5 Conclusion and Future Work

In this paper, we present MARAS, a robust and adaptive biologically-inspired mobile ad hoc network routing protocol. The next hop selection of this protocol uses the attractor selection mechanism inspired from cell biology. This protocol establishes the route reactively and maintains it by using the feedback packet for each delivered packet at the destination. The feedback packet evaluates the route that the data packet has taken and updates the activity at each node in the route, allowing the route to react to changes in the network without creating extra control overhead on changes. As a result, MARAS has a higher delivery efficiency than AODV because of its robustness to link failures, while creates lower overhead than AODV as it is adaptive to network dynamics. Moreover, according to the evaluation results, the overhead of MARAS remains almost constant and the delivery efficiency decreases slower than AODV when the node density increases, by which we conclude that MARAS is more scalable than AODV.

As a result from this study, an assumption has been made regarding the scalability advantage of maras that MARAS can avoid using the congested and unstable links. As future work, it would be interesting to see if MARAS can be used as a load balancing or traffic management protocol as the random walk mechanism has been found capable of such ability. At least, we could learn the limit of the traffic that MARAS can handle from this study, which could be useful for real world application.

Acknowledgment

This research was partly supported by the "Global COE (Centers of Excellence) Program" of the Ministry of Education, Culture, Sports, Science and Technology, Japan.

References

[1] M. S. Memon, M. Hashmani and N. A. Memon: "A review of uniqueness and variations in throughput due to performance parameters of simulators on manet routing protocols", Proceedings of the 7th WSEAS International Conference on Electronics, Hardware, Wireless and Optical Communications (EHAC), pp. 202–208 (2008).

[2] F. D. Rango, J.-C. Cano, M. Fotino, C. Calafate, P. Manzoni and S. Marano: "OLSR vs DSR: A comparative analysis of proactive and reactive mechanisms from an energetic point of view in wireless ad hoc networks", Computer Communications, **31**, 16, pp. 3843–3854 (2008).

[3] C. Perkins, E. Royer, S. Das and M. Marina: "Performance comparison of two on-demand routing protocols for ad hoc networks", Personal Communications, IEEE, **8**, 1, pp. 16–28 (2001).

[4] E. Borgia and F. Delmastro: "Effects of unstable links on aodv performance in real testbeds", EURASIP Journal on Wireless Communications and Networking, **2007**, 1, pp. 32–32 (2007).

[5] G. Di Caro, F. Ducatelle and L. Gambardella: "AntHocNet: An adaptive nature-inspired algorithm for routing in mobile ad hoc networks", European Transactions on Telecommunications, **16**, 5, pp. 443–455 (2005).

[6] H. Wedde, M. Farooq, T. Pannenbaecker, B. Vogel, C. Mueller, J. Meth and R. Jeruschkat: "BeeAdHoc: an energy efficient routing algorithm for mobile ad hoc networks inspired by bee behavior", Proceedings of the Genetic and Evolutionary Computation Conference (GECCO), pp. 153–160 (2005).

[7] N. Asvarujanon, K. Leibnitz, N. Wakamiya and M. Murata: "Extension and evaluation of biologically-inspired routing protocol for manets", IEICE technical report, **109**, 129, pp. 55–60 (2009).

[8] K. Leibnitz, N. Wakamiya and M. Murata: "Self-adaptive ad-hoc/sensor network routing with attractor-selection", Proceedings of IEEE Global Telecommunications Conference (GLOBECOM), pp. 1–5 (2006).

[9] K. Leibnitz, N. Wakamiya and M. Murata: "A bio-inspired robust routing protocol for mobile ad hoc networks", Proceedings of 16th International Conference on Computer Communications and Networks (ICCCN), pp. 321–326 (2007).

[10] A. Kashiwagi, I. Urabe, K. Kaneko and T. Yomo: "Adaptive response of a gene network to environmental changes by fitness-induced attractor selection", PLoS ONE, **1**, 1, p. e49 (2006).

[11] C. Perkins, E. Royer and S. Das: "Ad Hoc On-Demand Distance Vector Routing, draft-ietf-manet-aodv-08.txt", IETF MANET Working Group (2001).

[12] C. Perkins, E. Royer and S. Das: "Ad Hoc On-Demand Distance Vector Routing, draft-ietf-manet-aodv-09.txt", IETF MANET Working Group (2001).